WHAT IS CLAIMED IS:

1. A method of inter-thread communication in a multi-threaded computer comprises:
storing an inter-thread message in memory, the inter-thread message having a field for an address that indicates a location of data for a next thread to execute; and
writing to a self-destruct register after storing the message to indicate that a thread which stored the message in memory has completed execution, with the self-destruct register being cleared upon reading by the next thread.

2. The method of claim 1 wherein the inter-thread message field for an address provides an address of a register where the data for the next executing thread is stored.

3. The method of claim 1 wherein writing to a self-destruct register further comprises:
setting at least one bit in the self-destruct register which corresponds to the thread which is writing to the self-destruct register.

4. The method of claim 1 wherein writing to a self-destruct register further comprises:
setting one bit in the self-destruct register which corresponds to the thread which is writing to the self-destruct register.

5. The method of claim 1 further comprising:
writing to the self-destruct register by a first thread;
reading from the self-destruct register by a second thread, where the reading further comprises:
reading bits, if any, that are set in the self-destruct register; and
clearing all of the bits of the self-destruct register.

6. The method of claim 5 further comprises:
reading the inter-thread message from the memory by a new thread, where the new thread is a thread other than the first thread; and
executing the new thread.

7. A hardware-based multi-threaded processor comprises:

a general purpose processor that coordinates system functions;

a plurality of microengines that support multiple thread execution;

a scratchpad memory for storing inter-thread messages where execution of a write to the scratchpad memory by a first thread causes an address to be stored as an inter-thread message which indicates a location of data for a new thread; and

a self-destruct register for indicating the execution status of threads where reading of the self-destruct register clears all of the bits of the self-destruct register.

8. The processor of claim 7 wherein the plurality of microengines further comprise:

a register stack wherein execution of the write to the scratchpad memory by the first thread causes a register address of the register referenced by the write to be stored as the inter-thread message, with the register address indicating the register stack location of data for the new thread.

9. The processor of claim 7 wherein execution of a write to the self-destruct register by a thread causes at least one bit to be set in the self-destruct register, the bit being set corresponding to the thread which executed the write to the self-destruct register.

10. The processor of claim 7 wherein execution of a write to the self-destruct register by a thread causes one bit to be set in the self-destruct register, the bit set corresponding to the first thread which executed the write to the self-destruct register.

12. The processor of claim 11 wherein the read from the self-destruct register by the thread causes execution of a new thread for each bit that is set, if any, in the self-destruct register.

13. A computer program product residing on a computer readable medium causing a processor to perform a function comprises instructions causing the processor to:

store an inter-thread message in memory; and

set at least one bit in a self-destruct register.

14. The computer program of claim 13 further comprises instruction causing the processor to:

store a register address as the inter-thread message in memory.

15. The computer program of claim 13 further comprises instructions causing the processor to:

read the contents of the self-destruct register, the read also clearing the self-destruct register; and

execute a new thread according to any bits set in the self-destruct register.